



US Army Corps
of Engineers

The Hydrologic
Engineering Center

(4)

DTIC FILE COPY

AD-A198 748

MULTI-PURPOSE, MULTI-RESERVOIR SIMULATION ON A PC

DTIC
ELECTED
AUG 08 1988
S D
H

TECHNICAL PAPER NO. 119

AUGUST 1988

Approved for Public Release. Distribution is Unlimited.

38 8 38 125

Papers in this series have resulted from technical activities of the Hydrologic Engineering Center. Versions of some of these have been published in technical journals or in conference proceedings. The purpose of this series is to make the information available for use in the Center's training program and for distribution within the Corps of Engineers.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The contents of this report are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such commercial products.

Multi-purpose, Multi-Reservoir Simulation on a PC¹

Bill S. Eichert², M. ASCE, and Carl Franke³

INTRODUCTION AND OVERVIEW

The methodology and difficulties in converting a large, general purpose, mainframe, batch oriented computer program (for reservoir simulation) to work effectively in the PC environment are described in this paper. A brief overview of the present capabilities of the general purpose reservoir simulation program (HEC-5), that works on mainframe and MS DOS compatible computers, is also presented.

Major aspects of the conversion to the PC environment are presented, including the transformation of large subroutines written in FORTRAN IV, to a modular style in FORTRAN-77. Experiences with various Fortran compilers and techniques for memory management (considering the limited memory space available on the PC) are also presented. Overlaying and memory caching, used as implementation techniques for the PC, are discussed. Discussion includes using extended memory on the PC, which allows the program to operate in a non-overlaid form at speeds comparable to the HARRIS-1000 super minicomputer. Results showing the performance of the program as related to the HARRIS-1000 and various PC's are shown.

The recent enhancements to provide interactive execution of various utility programs for HEC-5 are discussed. These programs include: checking input data (CKHEC5); interactive input preparation (INFIVE); input modification (MODFIV); handling program executions (MENU5); and interactive output displays from HEC-5. Special "user friendly" features depicting spatial displays of input data and simulation results are shown.

BASIC PROGRAM OVERVIEW

The program "Simulation of Flood Control and Conservation Systems" (abbreviated HEC-5) was initially developed in 1972 utilizing large storage, high speed mainframe computers. Later versions of the program have been issued about once per year since 1979. The first PC version was released in 1987. The April 1988 version of HEC-5 and the latest companion programs consist of the following number of subroutines and lines of source code (including about 60 percent comment cards).

¹ Presented at ASCE 3rd Water Resources Operations and Management Workshop in Ft. Collins, Colorado, 27 - 30 June 1988

² Director, Hydrologic Engineering Center (HEC), 609 Second Street, Davis, CA 95616

³ Franke Technical Services, Woodland, Calif.

	<u>Number of Subroutines</u>	Lines of code <u>Total</u>
HEC5A	169	86,000
HEC5B	30	10,000
CKHEC5	28	9,600
IN5	23	10,800
MOD5	47	25,000
MENU5	<u>10</u>	<u>1,600</u>
Total	307	143,000

The HEC-5 program was developed to operate for one or more reservoirs for flood control, hydropower and water supply for all computational time intervals from 1 minute to 1 month. The capabilities to simulate the system for flow sequences up to 2000 time periods in length and for chaining these sequences together (for the same or different computational time intervals) were developed. Optimization routines are available to size single reservoir conservation storage or yields for hydropower, water supply or irrigation or for combinations of these for routing intervals from 1 day to 1 month. For flood control evaluations, seven different hydrologic routing techniques can be utilized to account for the stream flow routing effects (attenuation and delay). The HEC-5 program is being used by over 20 Corps of Engineers offices and 20 non-Corps offices based on information from an October 1986 questionnaire mailed to all of the offices on HEC's mailing list of HEC-5 source code holders. The offices responding represent over 150 users who are using HEC-5 on the following computers: Harris, CDC, IBM, Honeywell, Amdahl, Vax and several others. The following types of studies are being performed by these offices each year: 25 flood control; 20 hydropower; 23 water supply; 7 real-time flood control; and 4 irrigation.

Documentation for the HEC-5 package of programs includes: user manuals; PC implementation guides; structural flow charts; and listings of program variable use by subroutine name, statement number and type of Fortran statement.

MODERNIZATION OF SOURCE CODE

The HEC-5 program has been developed over a 16 year period on six or more mainframe and mini-computers under many operating environments and Fortran compilers. The scope of the program has changed from a simple flood control system to a highly complex multi-purpose, multi-reservoir, multi-flood simulation tool.

Due to the importance of minimizing program support costs, reducing program malfunctions, and facilitating new improvements, a decision was made in 1984 to modernize the source code. This modernization effort greatly assisted in the eventual conversion to the PC. The modernization process involved the following general steps:

- a. Modularize - In order to obtain a modular, top down design for HEC-5, structural flow charts were developed and large

subroutines were divided into smaller logical subroutines. HEC-5 has always been composed of two parts: HEC5A, the simulation part; and HEC5B, the output and economic analysis part. The 12 major subroutines in HEC5A were converted into 102 new subroutines. In addition to physically making smaller subroutines, many comment cards were added to explain the program organization and to clarify the logic within the new subroutines. Also, an ongoing effort is being made to reduce the number of temporary variables and to give them more meaningful names.

b. *Comdecks - Source code which is common to many subroutines, such as PARAMETER statements and COMMON BLOCKS, have been placed into separate files. These files are added to the source code during an update process or by the compiler at compile time. This activity is accomplished on the PC by the INCLUDE statement, while the Harris uses an update process called Historian (Opcode, 1987) which uses *Comdecks to include these files. The *comdeck approach was used for PARAMETER statements as well as for subroutine COMMONS and for most subroutine arguments. Using this approach a new variable can be added to a subroutine argument string and this modification is propagated throughout the source code such that all calls to that subroutine reflect the modified argument string.

c. Dimension limits - HEC-5 contains about 250 arrays resulting in extensive computer memory requirements. Many of the arrays have the same dimension limits such as maximum number of control points, reservoirs, and time periods. Since some computer systems may require that these limits be changed, it was desirable to use PARAMETER statements in the source code for 36 parameter variables so that the dimension limits can be changed easily. Limits of all logic loops dealing with these arrays are also keyed into these parameter variables. The PARAMETER statements are now all contained in a single *comdeck file which is brought into the subroutine during either the update or compilation phase. Thus, by editing one number in the PARAMETER statement and recompiling routines that contain the edited parameter statement, most key dimension limits can be easily changed.

d. COMMON Blocks - Prior to 1982, most variables were contained in one of several large COMMON blocks that were included in almost all subroutines. A major effort was made in 1982 to revise the COMMON blocks and to eliminate the major common blocks in all but a few major subroutines.

e. Centralize and Simplify Key Logic - Several new subroutines were developed in an attempt to centralize all logic dealing with certain functions in order to simplify the program logic. Several key routines were also rewritten because the logic had become too difficult to understand based on years of new improvements and error corrections.

QUALITY
INSPECTED
2

By _____	
Distribution/ _____	
Availability Codes _____	
Dist	Avail and/or Special
A-1	

f. Trace Output - Diagnostic output of intermediate results showing internal computations within the various subroutines is an important part of HEC-5. The modernization activity enhanced this output by including variable name labels above all numeric values along with subroutine name and statement number where the output originated.

g. FORTRAN-77 - In order to make the HEC-5 FORTRAN program easily transportable from one computer system and compiler to another, the source code was converted and updated to meet the ANSI FORTRAN-77 standard (ANSI, 1978). A major portion of the conversion process was eliminating the usage of hollerith variables and constants in the source code and replacing that usage with the FORTRAN-77 supported character variables.

Once the program was converted to character usage, the remainder of the effort centered around elimination of compiler specific code and replacing it with its FORTRAN-77 counterpart. In the case of the HARRIS compiler, INTEGER*6, and SPECIAL COMMON were reserved FORTRAN words which had no meaning to other compilers. These were removed from the source code.

In addition to changes required for compatibility with FORTRAN-77, other changes which used FORTRAN-77's more powerful features were also implemented, such as the PARAMETER statement previously mentioned. The FORTRAN IF-ELSE-ENDIF is another example. When source code was being modified, this construct was used to replace many lines of code which were dependent on the same conditional result.

CONVERSION OF HEC-5 TO THE PC

During the initial stages of adapting HEC-5 to execute in the MS-DOS (Microsoft, 1983) PC environment, several PC-based FORTRAN compilers were evaluated. By choosing the most appropriate compiler, it was hoped that the task of creating a HEC-5 executable for the PC could be accomplished as quickly and efficiently as possible. At the time (early 1987), four well known FORTRAN compilers were available for use with MS-DOS. They included Microsoft FORTRAN (Microsoft, 1985) (Version 3.3), Ryan-McFarland FORTRAN (Version 2.1) (Ryan, 1986), Lahey FORTRAN (Version 2.2) (Lahey, 1988), and Digital Research FORTRAN (Digital, 1986). Each of these compilers showed strengths and weaknesses when used for the initial compilation of the September 1986 HARRIS version of HEC-5. Since HEC-5 had just been converted to ANSI FORTRAN-77 standards, all aspects of the program complied with the full implementation of the ANSI FORTRAN-77 standard. However, it did not adhere to the specifications of the more restrictive subset of the FORTRAN-77 standard. This meant that a compiler which was a full implementation of ANSI standard would be preferred to an implementation of the subset. Several compilers were unable to handle subroutines that contained over 64 formal arguments. At this point in the development of the program, HEC-5 had many large subroutines with this characteristic (this was changed in 1988). If the compiler chosen could

handle large numbers of arguments, less time could be spent converting the program to an acceptable form for compilation.

The Lahey compiler was chosen because it alone had the full implementation of the ANSI FORTRAN-77 standard and could accommodate the large number of subroutine arguments. Also noted at the time, was that this compiler was compatible with an existing commercial overlay linker, Phoenix software's Plink86 (Phoenix, 1985) and could handle large common blocks greater than 64 thousand bytes (64 Kb), as well as large local arrays.

MEMORY CONSIDERATIONS ON THE PC

After successfully compiling the September 1986 version of HEC-5 using the Lahey compiler, the memory available for linking and execution of the program became the next important consideration. Attempting to run a program in the MS-DOS operating system on an 8088 based microprocessor brings with it several storage and size constraints that must be satisfied in order for the program to be able to run. The first limit to notice is the 1 megabyte (Mb) or 1000 Kb maximum addressing available on the 8088 microprocessor. Unless the window that the final program executes in is 1 Mb or less, the linker will not be able to create an executable. The second and probably most well known constraint is the 640 Kb maximum that DOS provides for program execution. If the program execution window is less than 1 Mb but greater than the 640 Kb limit, then the microprocessor will be able to run the executable, but will be prevented from doing so by DOS. The September 1986 version of HEC-5, if created as one large executable, would require 1.2 Mb of memory to execute. This is not only over the 640 Kb required by DOS, but also over the 1 Mb addressing space of the 8088. Clearly, something had to be done about the size of the executable in order to allow the program to operate on an 8088 based machine using MS-DOS.

Two steps were taken to deal with minimizing the amount of memory required for running the program. First, dimensional limits of large arrays and common blocks were reduced, allowing for creation of a smaller executable. The dimension limits were reduced for number of time periods (from 2,000 to 200), number of reservoirs (from 20 to 15) and number of control points (from 35 to 15). Secondly, the program was overlayed so that the program executable could reside in DOS's 640 Kb window.

PROGRAM OVERLAYING

One of the criteria used in choosing the proper FORTRAN compiler was that it be compatible with a linker that could do program overlaying. Because Phoenix Software's Plink86 linkage editor was not only capable of doing overlaying, but was also compatible with the Lahey FORTRAN, it was chosen to link the HEC-5 program. After a brief look at various hardware and software configurations of PC's at the Hydrologic Engineering Center, a target size for the HEC-5A executable of 550 Kb was chosen. This memory size would require the user to have the full 640 Kb complement of memory available in the PC, but still leave a little memory for system device

drivers and small ram resident utility programs.

The design of an overlay scheme could now be undertaken. Using the map feature of Plink86, memory requirements for each of the subroutine modules was determined. This information, coupled with the program design, was used to create an overlay scheme which was as efficient as possible, yet reduced the size of the resulting executable to the 550 Kb target size. Subroutines overlaid using this scheme were required to have all ancestor routines (those routines calling the subroutine) in memory with the overlay, further complicating the overlay structure. Since the program was already divided into three distinct sections, pre-operation, operation, and post-operation routines, this structure was also incorporated into the overlay design. The diagram of the resulting overlay map, showing the overlay groups, can be seen in figure 1. It should be noted that well over half of the overlay storage requirements are devoted to the common block storage area.

INITIAL PC PERFORMANCE

With the development of the September 1986 PC version of HEC-5, the program performance of HEC-5 could now be evaluated in the MS-DOS environment. The first machine used for testing the program was an Epson Equity III 'AT' compatible with a 80286 microprocessor. The machine was equipped with 640 Kb ram, a 1.2 Mb floppy drive, a 360 Kb floppy drive and a 80287 math coprocessor. Although all 10 standard test data sets executed properly, the process was much slower than comparable testing on the HARRIS 1000. The 10 standard tests which took about 8 min to execute on the HARRIS, took about 5 3/4 hrs to complete on the 8 MHz Epson computer (see figure 2). The same tests run on a IBM XT 8088 microcomputer took approximately 10 1/4 hrs. Clearly, the performance on the PC would have to be enhanced in order for the PC to become a viable alternative to operating the program on the HARRIS 1000 super mini computer.

OVERLAY MEMORY CACHING

Since code developed for the PC was heavily overlaid and required many relatively slow accesses to the hard disk to perform each overlay, attention was next focused on ways to improve program performance as it related to the overlays, for those users who have or were willing to buy extra memory. Either fewer overlays had to be implemented or the overlay process had to be done faster. Since extensive overlaying was required just to create the executable, speeding the overlay process was investigated. Plink86 had just the feature to accomplish this task. The overlay memory caching option allowed use of the same overlay scheme, but enabled the overlay process to occur at a much faster rate. Instead of overlaying subroutines from the slow hard disk, this option enabled overlaying from extended memory, if available. As the program is loaded into memory, the overlays are copied into 'high' memory. As each overlay is needed, it is copied from extended memory down to the 640Kb DOS environment where it is used. Program execution does not have to wait for slow disk accesses each time a overlay is required.

After the addition of 1 Mb of extended memory (costing approximately \$400) to accommodate the large number of overlays, the program was relinked on the Epson computer with the overlay cache option. Subsequent evaluation of program performance, showed that the overlay caching feature improved execution speeds by nearly a factor of four. The 10 standard tests which took 5 3/4 hrs to execute without overlay caching, now could execute in 1 1/2 hrs (see figure 2).

HEC-5 EXTENDED MEMORY OPERATION

As the experience of executing HEC-5 on the PC accumulated, it became increasingly apparent that mainframe performance of the program, both in terms of implementation as well as execution time could be realized on the PC. The decision was made to further improve the performance of the program in the PC environment. As a next step, the overlay cache version of the program was transported to a newer, faster PC. The new machine was a COMPAQ 386 with a 16MHz clock rate and a 80387 math coprocessor. Again performance of the program improved, this time due to the faster raw processing speed of the 80386. The 10 standard tests could now be executed in approximately 32 min with the overlay caching.

At about the same time, Lahey announced the distribution of an extended memory FORTRAN compiler for use on 80286 and 80386 based microcomputers. This compiler would allow execution of large FORTRAN programs in extended memory without the use of overlays. This announcement had 2 implications on the development of HEC-5 on the PC. First, since overlays would not be necessary, program performance should increase once again. Secondly, since program size was dependent on the amount of extended memory available and not on the 640 Kb limit imposed by DOS, the entire mainframe version of HEC-5 could be implemented on the PC without compromising array sizes to obtain a smaller executable.

Coincident with the development of HEC-5 on the PC were several major enhancements to the mainframe version of HEC-5. The most notable of these improvements was the addition of the interactive display menu allowing users to view intermediate output produced by the program. When the new version of the program became available in April 1988, work was started immediately to implement the new full mainframe version on the PC using the extended memory FORTRAN compiler from Lahey.

The extended memory FORTRAN compiler system allowed for greater capabilities of the program in the PC environment. Not only did it lift the 640 Kb memory requirement of DOS, but it did so without abandoning DOS. The extended memory compiler and OS/286 operating system, (A.I. Architects, 1988) take advantage of the protected mode operation of the 80286 microprocessor while maintaining DOS compatibility. The OS/286 acts as a supplement to DOS rather than replacing it. This means that all the DOS compatible utility programs are still available to the user. OS/286 is only invoked when needed by the FORTRAN executable at run time of the program. Upon completion of the program, control is again returned back to DOS.

CURRENT STATUS OF HEC-5 ON THE PC

Currently, the April 1988 mainframe version of HEC-5 is running on the COMPAQ 368 PC. It is a full implementation of the mainframe version of the program, including the interactive capabilities of the menu system. As on the HARRIS version, the HEC data storage system (DSS) is also available for obtaining graphical displays of both input and computed time series data. The program currently requires approximately 2.3 Mb extended memory (plus 500 Kb for the operating system), for use with either a 80286 or 80386 based microprocessor. It executes the 10 standard tests in approximately 18 min. which is just over twice the execution time on the HARRIS 1000 computer.

Work is currently under way to interface new capabilities to the PC version of the program by the addition of water quality subroutines which will work in conjunction with the quantity subroutines of HEC-5A.

FUTURE PC DIRECTIONS

Due to the success of adapting HEC-5 to the PC environment, more program development is being planned for the PC. The next step in the process of improving performance and capabilities of the program is the conversion of the program from 16 bit operation to 32 bit operation. By recompiling the program, with Lahey 32-bit extended memory FORTRAN compiler and linking with the new A.I. Architects OS/386 operating system, this conversion can be accomplished. With this conversion, HEC-5 will be able to access the full power of the 80386 and 80387 chips. Data transfers which currently occur in 2 16-bit transfers would be replaced by 1 32-bit data transfer yielding even further improvement in performance. Use of the 32-bit FORTRAN compiler and OS/386 operating system should allow the PC to meet or surpass the performance of the HARRIS 1000 computer.

In addition to a new 32-bit extended memory version of HEC-5, future plans also include implementation of the program in the new OS/2 PC operating system. With the arrival of Microsoft Fortran (version 4.1) and OS/2 (version 1.0) development for this new environment is ready to proceed.

For users of 8088 microcomputers interested in running the latest (April 1988) version of HEC-5, and willing to sacrifice performance, an overlayed, 16 bit version of the program may also be developed. This version would operate slower than the extended memory version of the program, but would not require extended memory or an 80286 or 80386 microprocessor.

ENHANCING BATCH PROGRAMS FOR PC ENVIRONMENT

In early 1987, versions of HECSA (simulation module) and HECSB (output module) worked for the first time on the PC in the batch mode. While these programs have the same "user friendliness" of the mainframe computers, the PC environment offered, even demanded, more "user friendliness". Therefore, several utility programs were developed (and

a few existing ones were modified) to enhance the "user friendliness" on both the PC and mainframe.

The first program in this category which was assembled for this user friendly environment was the CKHEC5 (CK5) program. This batch program, developed in 1985, reads the HEC-5 input data file describing the user's system and checks the data to see if it was prepared following the rules in the users manual. Possible error messages are printed if potential difficulties are encountered.

To assist the user in deciding what input data is needed by HEC-5 for particular reservoir systems, a new program called INFIVE (IN5) was developed in 1985 and 1986. This interactive input program asks the user a number of questions concerning the physical and operational makeup of the reservoir system to be modeled. At the conclusion of the 10-15 minute session, a skeleton data file in the form of 80 column card images has been developed to make the HEC-5 simulation. However, additional data must be supplied before a simulation can be done. Inputting values for specific data fields is accomplished by using either a text editor and the Exhibit 8 of the HEC-5 User Manual (Hydrologic Engineering Center, 1988) or by using an HEC editor called COED (Hydrologic Engineering Center, 1987). In 1987, COED was modified to provide help files for various HEC programs, including HEC-5. The COED help files provide either variable names or variable definitions for each data item as it is pointed to by the screen cursor.

In order to allow occasional users of HEC-5 to modify operational parameters during real-time flood control operations, a new interactive program called MODFIV (MOD5) was developed in 1986 - 1988. This MOD5 program uses menus (see figure 3 for an example) to display and change data representing all required cards and all operational parameters. At the conclusion of the interactive session, a new HEC-5 data file is available representing the modifications requested during the interactive session.

The remaining link in enhancing the user environment is to allow these different programs to be executed in a user friendly manner from a menu. Thus, a program called MENU5 was developed in 1987 and 1988 for the PC which will allow the following programs to be executed:

<u>NAME</u>	<u>PURPOSE</u>
CK5	Input Data Checking Program (CKHEC5)
IN5	Interactive Input Preparation Program (INFIVE)
MOD5	Interactive Input Modification Program (MODFIV)
COED	Interactive Editor
HEC5A	Simulation part of HEC-5 (HEC5A)
HEC5B	Output and Economic Summary part of HEC-5 (HEC5B)
DISPLAY	Graphical Data Display Utility (DISPLAY)

In addition to executing the above programs from a menu, MENU5 will also provide a convenient way to display or print output results, to transfer file names among programs, and to facilitate naming the input

file. For instance, if desired, a project name can be input and all files required by all programs will be automatically named. Also a list of all available HEC-5 input files can be used to make the file selection by positioning the screen cursor.

Enhancements recently completed include interactive output displays and interfacing with HEC's Data Storage System (DSS). The April 1988 mainframe and PC versions of the HEC5A program can now be executed in either the batch or interactive modes. In the interactive mode a menu will be displayed on the screen after the simulation is complete (see figure 4). This menu will allow the user to look at various HEC-5 standard output tables or to create tables or color plots (see figure 5) of any time series data using results which are in memory. Thus, the operational results can be quickly inspected from the MENU without needing to display or print large volumes of output. Two useful new types of output developed are the schematic map of the basin, one to display input and the other to display output.

The HEC-DSS system (available both on mainframe and PC versions) is used for displaying time series results in a graphical form using the HEC program DSPLAY. In addition, any HEC-5 time series data can be read from DSS or stored into DSS. HEC5A and MOD5 currently work with DSS, while IN5 and CKHEC5 do not currently look for flow data stored in DSS.

CURRENT PC PERFORMANCE

The September 1986 and April 1988 versions of HEC-5 are presently on the PC. The 1988 source code for the PC and the mainframe are the same except for a few statements. Comparisons between the HARRIS 1000 and the PC versions are made in terms of maximum limits of key arrays, memory requirements, FORTRAN Compilers, and execution speeds (see figure 2).

Maximum Array Sizes: The maximum array sizes for key variables are as follows:

	Mainframe	1986 PC	1988 PC
Number of Reservoirs	20	15	20
Number of Control Points	35	15	35
Number of Time Periods	2,000	200	2,000
Number of Power Plants	12	9	12

Memory Requirements: The execution times shown in figure 2 are highly dependent on the following use of overlay, extended memory and memory caching for the various versions:

	Mainframe	1986 PC	1988 PC
Dynamic Time Series Data	YES	YES	YES
Overlay Used	NO	YES	NO
Extended Memory	NO	NO	YES
Overlay Caching	N/A	YES	NO

Compilers: Times shown are the clock times required to compile (all 169 subroutines) and link the HEC5A routines on the compilers currently in use with HEC-5.

Fortran-77 <u>Compiler</u>	Compile <u>Time (min)</u>	Link <u>Time (min)</u>
HARRIS VOS 6.1.0	15	2
LAHEY FORTRAN	25	15
LAHEY EXTENDED MEMORY FORTRAN	25	
15 MICROSOFT	45	—*

* D group size larger than 64 kb link not complete

REFERENCES

A.I. Architects, care of Lahey Computer Systems, Inc., 1988

American National Standards Institute, Inc., "American National Standard Programming Language, FORTRAN - ANSI X3.9-1978, "commonly known as FORTRAN 77, New York, NY 1978.

Digital Research Corporation, Pacific Grove, California, 1986.

Historian Plus, Opcode Inc., Austin, TX, February 1987.

Hydrologic Engineering Center, U.S. Army Corps of Engineers, Davis, Ca 95616, 1987, 1988

Lahey Computer Systems, Inc., P.O. Box 6091, Incline Village, NV 89450-6091, 1988.

Microsoft Corporation, "Microsoft Disk Operating System (MS-DOS) Version 2.1, "(and higher), Bellevue, WA, 1983.

Microsoft Corporation, "Microsoft FORTRAN Compiler for the MS-DOS Operating System, "Bellevue, WA, 1985.

Phoenix Software Associates, Ltd., "PTEL - PLINK86 plus, "Norwood, MA, October 1985.

Ryan-McFarland Corporation, "RM/FORTRAN Compiler for the MS-DOS Operating System, "Rolling Hills Estates, CA January 1986.

TRADEMARKS

This paper refers to the following trademarked products:

- OS/286, OS/386 are trademarks of A.I. Architects, Inc.
- Compaq 386 is a trademarks of Compaq Computer Corporation.
- MS DOS is a trademark of Microsoft, Inc.
- INTEL 8088, 8087, 80286, 80287, 80386, 80387 are trademarks of INTEL, Inc.
- Epson Equity III is a trademark of Seiko Epson Corporation
- QUADRAM board is a trademark of Quadram Corp.

HEC-5 OVERLAY STRUCTURE
September 1986 - PC Version

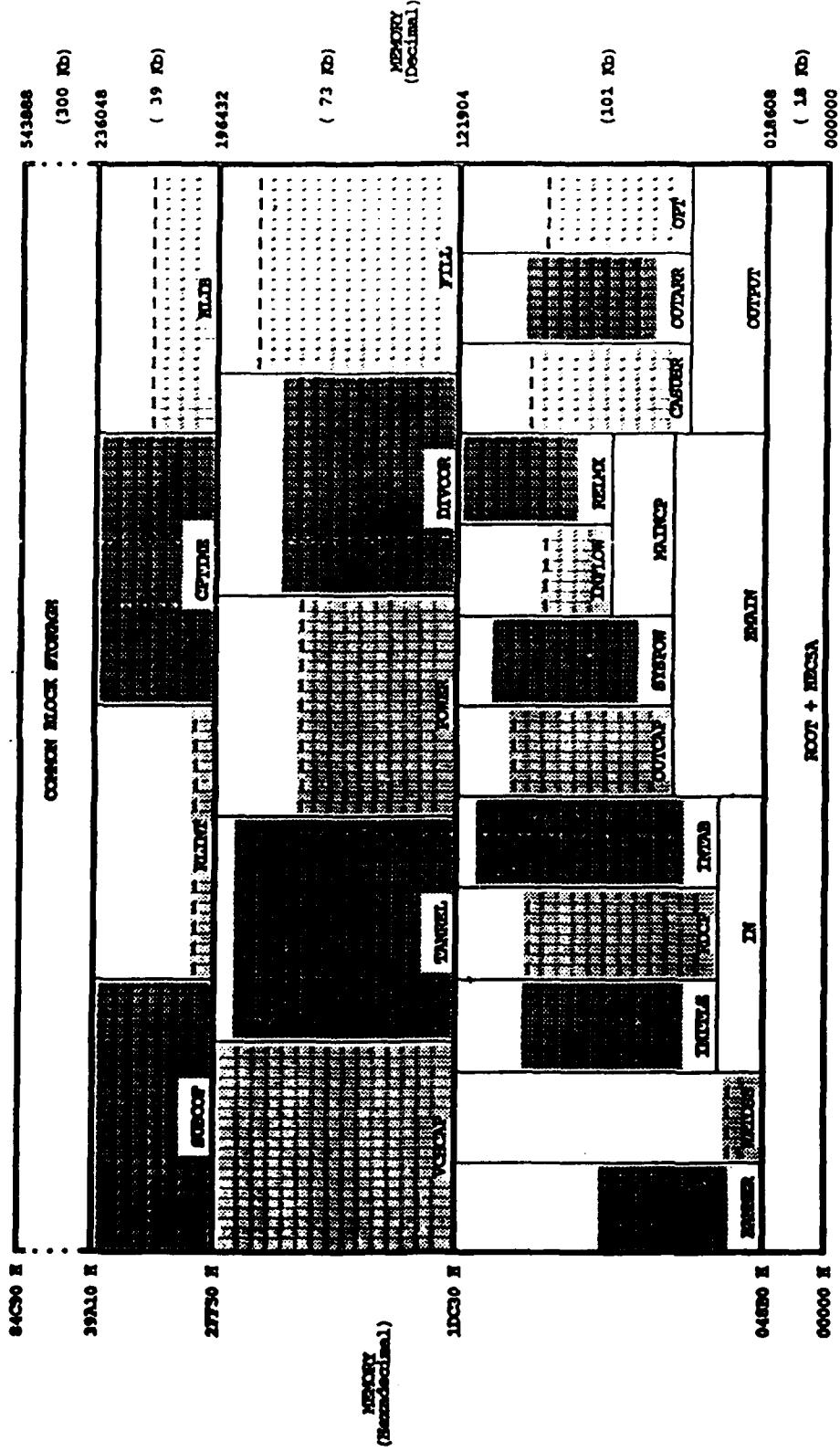


Figure #1

NEC-S EXECUTION TIMES
Micro and Miniature Computers

Date <u>2021</u> ¹	Job <u>1722</u>	Number of Res./C.P.s/Periods	Execution Time (Hr:Min:Sec)					
			PC/XT2	PC/AT3	Overloaded PC/AT4	PC/3865	PC/3866	Non-Overloaded PC/3867 Harris 1000 ⁸
Test 1	Flood Control	1	4	10	9:44	5:30	1:36	1:57
Test 2	Flood Control	2	3	14	12:24	7:02	1:48	2:26
Test 3	Flood Control	3	4	17	19:52	11:00	2:40	3:56
Test 4	Hydroponer	1	2	150	1:18:49	44:01	11:58	17:47
Test 5	Hydroponer	1	2	120	1:10:48	39:49	10:37	16:24
Test 6	Water Supply	3	6	48	32:38	18:12	4:43	6:57
Test 7	Optimization	1	2	480	2:56:02	1:38:14	26:52	46:29
Test 8	Hydroponer	4	5	35	1:08:09	38:09	11:45	14:08
Test 9	Hydroponer	3	4	58	52:02	29:41	9:20	12:43
Test 10	Flood Control	3	4	108	1:33:14	52:02	10:48	21:22
Total Time						3:46	1:51	0:48
						16:17	32:26	8:08

Data sets are shown in "NEC-5, Users Manual, March 1982."

2100 SCART counter with an 8087 math coprocessor. 2.1 MS-DOS operating system. 4.77 MHz clock rate and 660KB memory.

THE JOURNAL OF CLIMATE

except with a 100 GIANAN board creating approximately 24001 contained as extended memory.

THE JOURNAL OF CLIMATE

卷之三

Operating system, extended memory compiler.

8. What is 1000 super mini computer (no other logic than memory).

Figure 2

MOD5 MENU FOR FLOOD CONTROL GOALS

T1 Scioto River Basin

		(Menu N-F) FLOOD CONTROL GOALS:					
COLUMN -		0	1	2	3	4	
	C P NAME	MAX FLOW	R/C-RIS	R/C-FAL	HRS-FCST		
255R	DDOD2	5000.	1000.	1000.	0.		
268	WOOE2	5000.	0.	0.	0.		
275	CLSF4	20000.	0.	0.	0.		
	---288R	ACSE3	1800.	300.	1000.	0.	
	---290	CLAF3	2100.	0.	0.	0.	
	---284R	HOWE3	4321.	50000.	50000.	0.	
307	---295	REWF3	9120.	0.	0.	0.	
		CISG3	15700.	0.	0.	0.	

Press ENTER key for CURRENT Menu
or to EDIT values, Type Location NUMBER, Column NUMBER (1- 4),
AND "New" VALUE; then Press ENTER key

Figure 3

HEC-5 EXECUTION IN INTERACTIVE MODE

WELCOME to Program HEC-5 and the "Optional" Capability
to "Interactively" Display
the OUTPUT RESULTS

Messages will be displayed during the program's simulation
to inform you of the progress of the program. When the
simulation has completed, you will then be given instructions
on how to display the results.

Reservoir Simulation has begun:

HEC-5 Simulation Results have been computed for:

Period = 1 of 81 18 JAN 1959 (2400 hours)
|>-----|

Period = 40 of 81 23 JAN 1959 (2100 hours)
|----->-----|

Period = 81 of 81 28 JAN 1959 (2100 hours)
|----->|

*** HEC-5 OUTPUT MENU ***

U - USER Designed Output

S - SUMMARY of MAXIMUM Events

R - REGULATED Flows by Time Period

E - ERROR Check Messages

C - CASE Codes for Reservoir Decisions

T - TERMINATE "Interactive" Output Display
(Continue with H5A)

Q - QUIT Program H5A

Enter letter to select option:

Figure 4

HEC-5 PLOTS OF OPERATION RESULTS

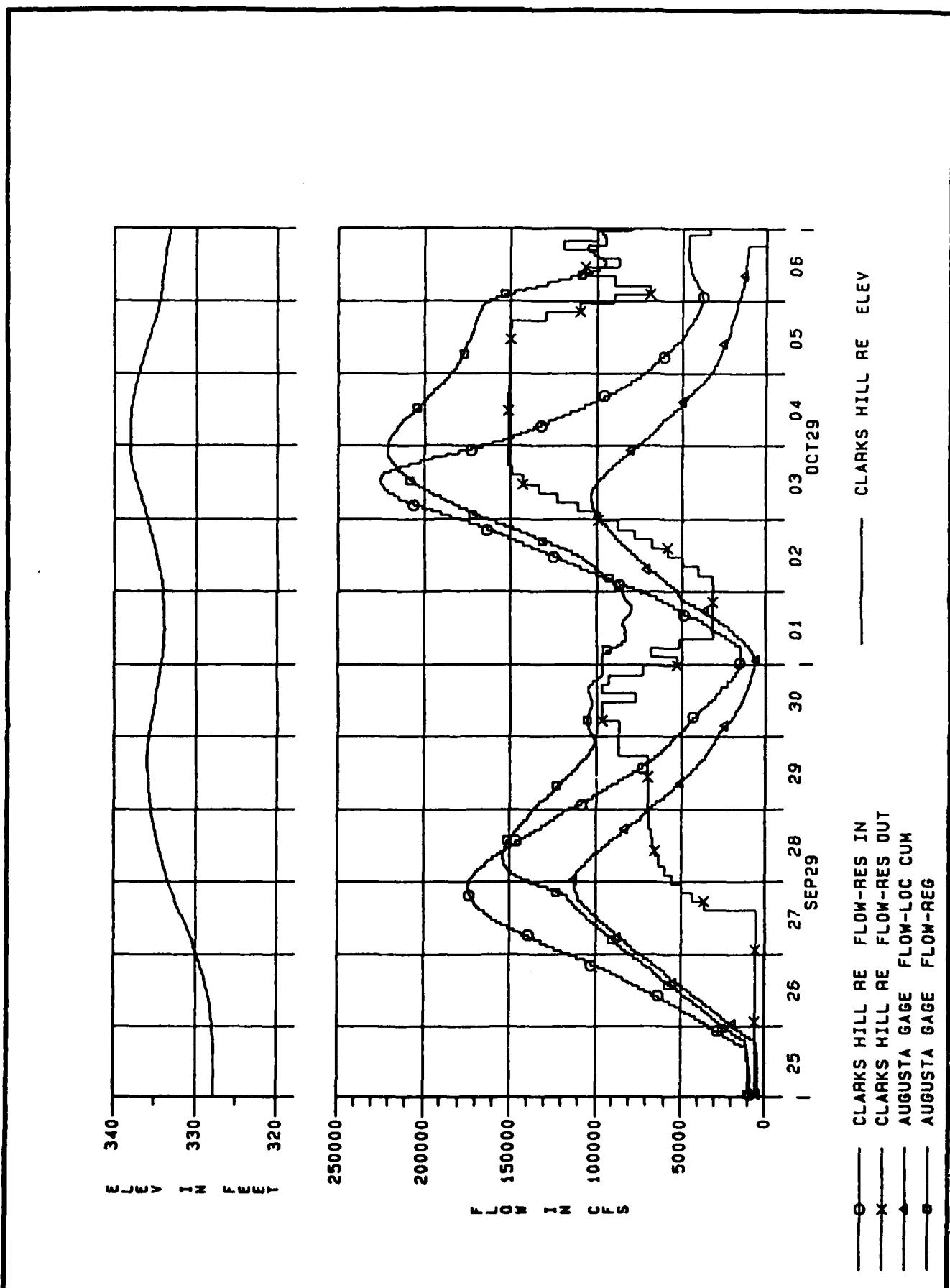


Figure 5

REPORT DOCUMENTATION PAGE			
1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED		1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION/AVAILABILITY OF REPORT	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE			
4. PERFORMING ORGANIZATION REPORT NUMBER(S) TECHNICAL PAPER NO. 119		5. MONITORING ORGANIZATION REPORT NUMBER(S)	
6a. NAME OF PERFORMING ORGANIZATION US ARMY CORPS OF ENGINEERS HYDROLOGIC ENGINEERING CENTER	6b. OFFICE SYMBOL (if applicable) CEWRC-HEC	7a. NAME OF MONITORING ORGANIZATION	
6c. ADDRESS (City, State, and ZIP Code) 609 SECOND STREET DAVIS, CA 95616		7b. ADDRESS (City, State, and ZIP Code)	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION WATER RESOURCES SUPPORT CENTER	8b. OFFICE SYMBOL (if applicable) CEWRC	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c. ADDRESS (City, State, and ZIP Code) CASEY BUILDING #2594 FT. BELVOIR, VA 22060-5586		10. SOURCE OF FUNDING NUMBERS	
11. TITLE (Include Security Classification) MULTI-PURPOSE, MULTI-RESERVOIR SIMULATION ON A PC	(UNCLASSIFIED)		
12. PERSONAL AUTHOR(S) BILL S. EICHERT, CARL FRANKE			
13a. TYPE OF REPORT TECHNICAL PAPER	13b. TIME COVERED FROM _____ TO _____	14. DATE OF REPORT (Year, Month, Day) 1988, AUGUST	15. PAGE COUNT 16
16. SUPPLEMENTARY NOTATION			
17. COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number) Simulation, Personal Computer, Reservoir System, Flood Control, Hydropower, Water Supply, Computer Programs, Multi-Purpose, Multi-Reservoir, FORTRAN. (T-E-S) ←	
19. ABSTRACT (Continue on reverse if necessary and identify by block number) The methodology and difficulties in converting a large, general purpose, mainframe, batch oriented computer program (for reservoir simulation) to work effectively in the PC environment are described in this paper. A brief overview of the present capabilities of the general purpose reservoir simulation program (HEC-5), that works on mainframe and MS DOS compatible computers, is also presented. Key words			
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS		21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL BILL S. EICHERT		22b. TELEPHONE (Include Area Code) (916) 551-1748	22c. OFFICE SYMBOL CEWRC-HEC

TECHNICAL PAPER SERIES
(\$2.00 per paper)

- TP-1 Use of Interrelated Records to Simulate Streamflow
TP-2 Optimization Techniques for Hydrologic Engineering
TP-3 Methods of Determination of Safe Yield and Compensation Water from Storage Reservoirs
TP-4 Functional Evaluation of a Water Resources System
TP-5 Streamflow Synthesis for Ungaged Rivers
TP-6 Simulation of Daily Streamflow
TP-7 Pilot Study for Storage Requirements for Low Flow Augmentation
TP-8 Worth of Streamflow Data for Project Design - A Pilot Study
TP-9 Economic Evaluation of Reservoir System Accomplishments
TP-10 Hydrologic Simulation in Water-Yield Analysis
TP-11 Survey of Programs for Water Surface Profiles
TP-12 Hypothetical Flood Computation for a Stream System
TP-13 Maximum Utilization of Scarce Data in Hydrologic Design
TP-14 Techniques for Evaluating Long-Term Reservoir Yields
TP-15 Hydrostatistics - Principles of Application
TP-16 A Hydrologic Water Resource System Modeling Techniques
TP-17 Hydrologic Engineering Techniques for Regional Water Resources Planning
TP-18 Estimating Monthly Streamflows Within a Region
TP-19 Suspended Sediment Discharge in Streams
TP-20 Computer Determination of Flow Through Bridges
TP-21 An Approach to Reservoir Temperature Analysis
TP-22 A Finite Difference Method for Analyzing Liquid Flow in Variably Saturated Porous Media
TP-23 Uses of Simulation in River Basin Planning
TP-24 Hydropower Analysis in Reservoir Systems
TP-25 Status of Water Resource Systems Analysis
TP-26 System Relationships for Panama Canal Water Supply
TP-27 System Analysis of the Panama Canal Water Supply
TP-28 Digital Simulation of an Existing Water Resources System
TP-29 Computer Applications in Continuing Education
TP-30 Drought Severity and Water Supply Dependability.
TP-31 Development of System Operation Rules for an Existing System by Simulation
TP-32 Alternative Approaches to Water Resource System Simulation
TP-33 System Simulation for Integrated Use of Hydroelectric and Thermal Power Generation
TP-34 Optimizing Flood Control Allocation for a Multipurpose Reservoir
TP-35 Computer Models for Rainfall-Runoff and River Hydraulic Analysis
TP-36 Evaluation of Drought Effects at Lake Atitlan
TP-37 Downstream Effects of the Levee Overtopping at Wilkes-Barre, PA, During Tropical Storm Agnes
TP-38 Water Quality Evaluation of Aquatic Systems
TP-39 A Method for Analyzing Effects of Dam Failures in Design Studies
TP-40 Storm Drainage and Urban Region Flood Control Planning
TP-41 HEC-5C, A Simulation Model for System Formulation and Evaluation
TP-42 Optimal Sizing of Urban Flood Control Systems
TP-43 Hydrologic and Economic Simulation of Flood Control Aspects of Water Resources Systems
TP-44 Sizing Flood Control Reservoir Systems by Systems Analysis
TP-45 Techniques for Real-Time Operation of Flood Control Reservoirs in the Merrimack River Basin
TP-46 Spatial Data Analysis of Nonstructural Measures
TP-47 Comprehensive Flood Plain Studies Using Spatial Data Management Techniques
TP-48 Direct Runoff Hydrograph Parameters Versus Urbanization
TP-49 Experience of HEC in Disseminating Information on Hydrological Models
TP-50 Effects of Dam Removal: An Approach to Sedimentation
TP-51 Design of Flood Control Improvements by Systems Analysis: A Case Study
TP-52 Potential Use of Digital Computer Ground Water Models
TP-53 Development of Generalized Free Surface Flow Models Using Finite Element Techniques
TP-54 Adjustment of Peak Discharge Rates for Urbanization
TP-55 The Development and Servicing of Spatial Data Management Techniques in the Corps of Engineers
TP-56 Experiences of the Hydrologic Engineering Center in Maintaining Widely Used Hydrologic and Water Resource Computer Models
TP-57 Flood Damage Assessments Using Spatial Data Management Techniques
TP-58 A Model for Evaluating Runoff-Quality in Metropolitan Master Planning

- TP-59 Testing of Several Runoff Models on an Urban Watershed
- TP-60 Operational Simulation of a Reservoir System with Pumped Storage
- TP-61 Technical Factors in Small Hydropower Planning
- TP-62 Flood Hydrograph and Peak Flow Frequency Analysis
- TP-63 HEC Contribution to Reservoir System Operation
- TP-64 Determining Peak-Discharge Frequencies in an Urbanizing Watershed: A Case Study
- TP-65 Feasibility Analysis in Small Hydropower Planning
- TP-66 Reservoir Storage Determination by Computer Simulation of Flood Control and Conservation Systems
- TP-67 Hydrologic Land Use Classification Using LANDSAT
- TP-68 Interactive Nonstructural Flood-Control Planning
- TP-69 Critical Water Surface by Minimum Specific Energy Using the Parabolic Method
- TP-70 Corps of Engineers Experience with Automatic Calibration of a Precipitation-Runoff Model
- TP-71 Determination of Land Use from Satellite Imagery for Input to Hydrologic Models
- TP-72 Application of the Finite Element Method to Vertically Stratified Hydrodynamic Flow and Water Quality
- TP-73 Flood Mitigation Planning Using HEC-SAM
- TP-74 Hydrographs by Single Linear Reservoir Model
- TP-75 HEC Activities in Reservoir Analysis
- TP-76 Institutional Support of Water Resource Models
- TP-77 Investigation of Soil Conservation Service Urban Hydrology Techniques
- TP-78 Potential for Increasing the Output of Existing Hydroelectric Plants
- TP-79 Potential Energy and Capacity Gains from Flood Control Storage Reallocation at Existing U. S. Hydropower Reservoirs
- TP-80 Use of Non-Sequential Techniques in the Analysis of Power Potential at Storage Projects
- TP-81 Data Management Systems for Water Resources Planning
- TP-82 The New HEC-1 Flood Hydrograph Package
- TP-83 River and Reservoir Systems Water Quality Modeling Capability
- TP-84 Generalized Real-Time Flood Control System Model
- TP-85 Operation Policy Analysis: Sam Rayburn Reservoir
- TP-86 Training the Practitioner: The Hydrologic Engineering Center Program
- TP-87 Documentation Needs for Water Resources Models
- TP-88 Reservoir System Regulation for Water Quality Control
- TP-89 A Software System to Aid in Making Real-Time Water Control Decisions
- TP-90 Calibration, Verification and Application of a Two-Dimensional Flow Model
- TP-91 HEC Software Development and Support
- TP-92 Hydrologic Engineering Center Planning Models
- TP-93 Flood Routing Through a Flat, Complex Flood Plain Using a One-Dimensional Unsteady Flow Computer Program
- TP-94 Dredged-Material Disposal Management Model
- TP-95 Infiltration and Soil Moisture Redistribution in HEC-1
- TP-96 The Hydrologic Engineering Center Experience in Nonstructural Planning
- TP-97 Prediction of the Effects of a Flood Control Project on a Meandering Stream
- TP-98 Evolution in Computer Programs Causes Evolution in Training Needs: The Hydrologic Engineering Center Experience
- TP-99 Reservoir System Analysis for Water Quality
- TP-100 Probable Maximum Flood Estimation - Eastern United States
- TP-101 Use of Computer Program HEC-5 for Water Supply Analysis
- TP-102 Role of Calibration in the Application of HEC-6
- TP-103 Engineering and Economic Considerations in Formulating
- TP-104 Modeling Water Resources Systems for Water Quality
- TP-105 Use of a Two-Dimensional Flow Model to Quantify Aquatic Habitat
- TP-106 Flood-Runoff Forecasting with HEC-1F
- TP-107 Dredged-Material Disposal System Capacity Expansion
- TP-108 Role of Small Computers in Two-Dimensional Flow Modeling
- TP-109 One-Dimensional Model For Mud Flows
- TP-110 Subdivision Froude Number
- TP-111 HEC-5Q: System Water Quality Modeling
- TP-112 New Developments in HEC Programs for Flood Control
- TP-113 Modeling and Managing Water Resource Systems for Water Quality
- TP-114 Accuracy of Computed Water Surface Profiles - Executive Summary
- TP-115 Application of Spatial-Data Management Techniques in Corps Planning
- TP-116 The HEC's Activities in Watershed Modeling
- TP-117 HEC-1 and HEC-2 Applications on the MicroComputer
- TP-118 Real-Time Snow Simulation Model for the Monongahela River Basin